Trust-based local and social recommendation

Simon Meyffret Lionel Médini Frédérique Laforest

University of Lyon, CNRS, France
INSA-Lyon, LIRIS, UMR5205, F-69621, France
Université Lyon 1, LIRIS, UMR5205, F-69622, France
simon.meyffret@liris.cnrs.fr

9 September 2012

Context

Collaborative Filtering Recommender Systems

- Data management
  - → Where are the data?
  - → Privacy
- Architecture
  - → Data decentralization
  - → P2P

Recommender System purely local

⇒ No global knowledge on ratings

⇒ User-centric data management

Context

Collaborative Filtering Recommender Systems

- Data management
  - → Where are the data?
  - → Privacy
- Architecture
  - → Data decentralization
  - → P2P

Recommender System purely local

⇒ No global knowledge on ratings

⇒ User-centric data management

Plan

Plan

Brief state of the art

Trust-based Recommender Systems

- MoleTrust [Massa2007]
  - → trust propagation
- RandomWalk [Jamali2009]
  - → purely local
- TrustWalker [Jamali2009]
  - → default scores based on item similarity

Brief state of the art

Trust-based Recommender Systems

- MoleTrust [Massa2007]
  → trust propagation
- RandomWalk [Jamali2009]
  → purely local
- TrustWalker [Jamali2009]
  → default scores based on item similarity

Brief state of the art

Trust-based Recommender Systems

- MoleTrust [Massa2007]
  → trust propagation
- RandomWalk [Jamali2009]
  → purely local
- TrustWalker [Jamali2009]
  → default scores based on item similarity

Plan

Communication

Solely based on links in the social network

- No new link in the social network
- Trust and similarity weight scores
- Only friends may share data

P2P style communication

- 1 peer = 1 user (*aka* actor)
- 1 link = 1 explicit social link

Communication

Solely based on links in the social network

- No new link in the social network
- Trust and similarity weight scores
- Only friends may share data

P2P style communication

- 1 peer = 1 user (*aka* actor)
- 1 link = 1 explicit social link

Proposition

Predict scores through the social network

- If there is a rating on the item, return it
- Otherwise ask friends
- Who will ask their friends
- ...

$\rightarrow$ Up to depth k

~~Trust propagation~~ $\Rightarrow$ Score propagation

Proposition

Predict scores through the social network

- If there is a rating on the item, return it
- Otherwise ask friends
- Who will ask their friends
- ...

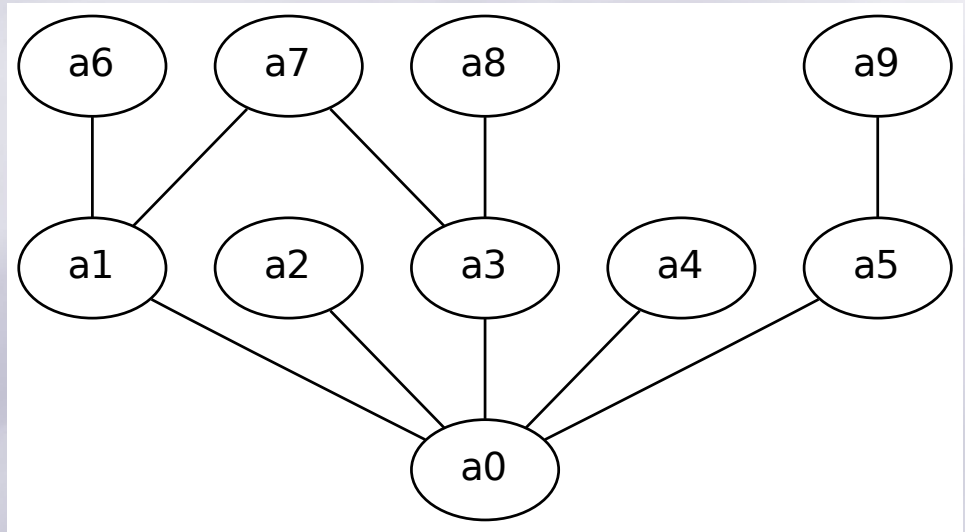
$\rightarrow$ Up to depth k

~~Trust propagation~~ $\Rightarrow$ Score propagation

Example

a0	a1	a2	a3	a4	a5	a6	a7	a8	a9
X	0.2	$\perp$	$\perp$	0.8	$\perp$	0.6	0.9	0.5	0.1

(a) Actors' ratings on item i_0



(b) Social network

Figure : Social network and ratings example centered around a_0

Score propagation example

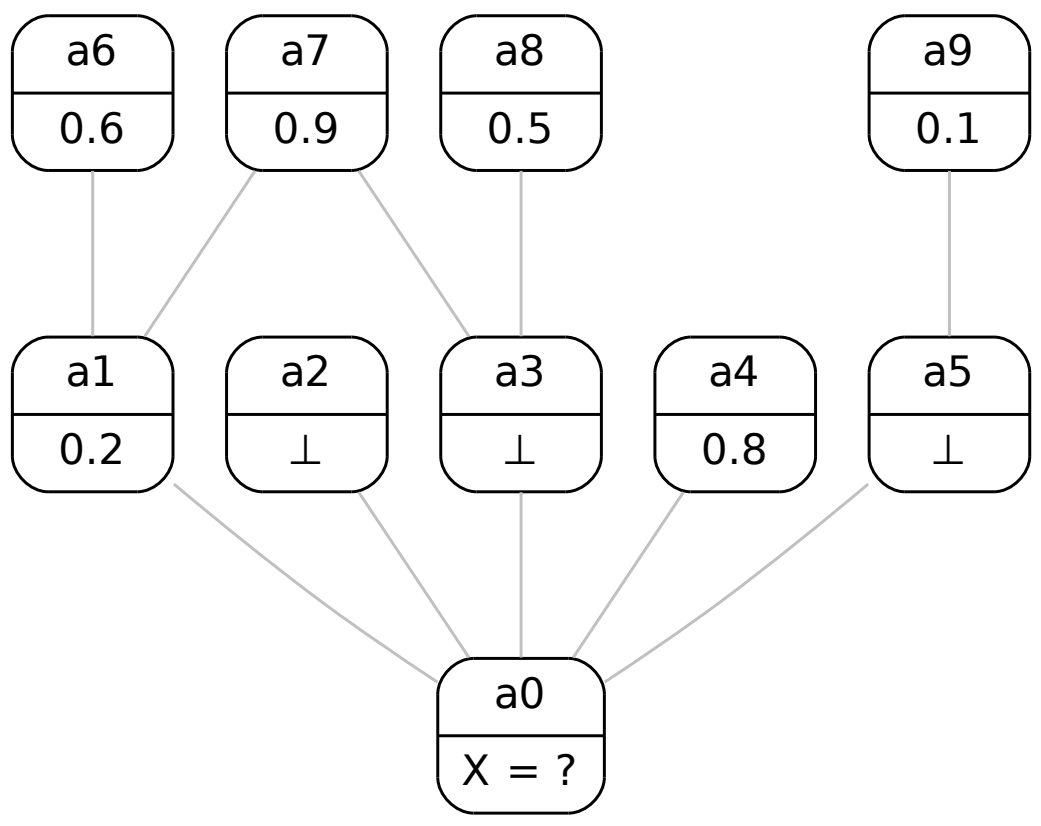


Figure : $k-$Depth Social Scoring Example with $k = 2$

Score propagation example

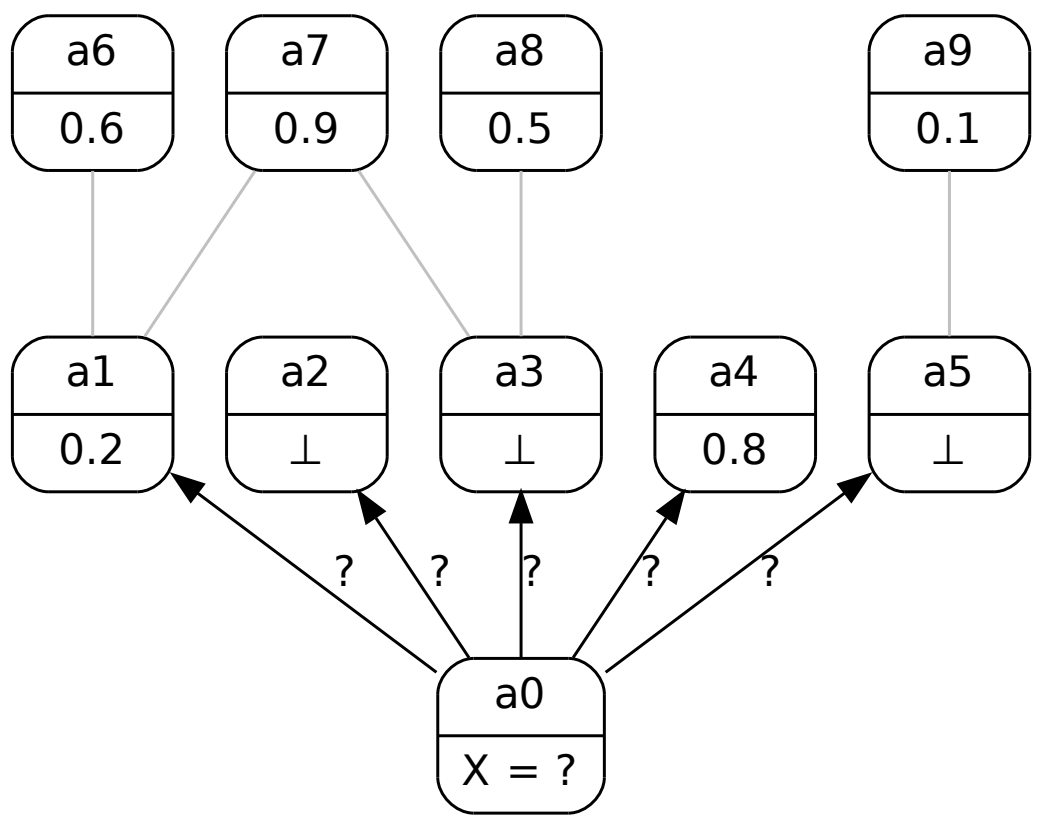


Figure : $k-$Depth Social Scoring Example with $k = 2$

Score propagation example

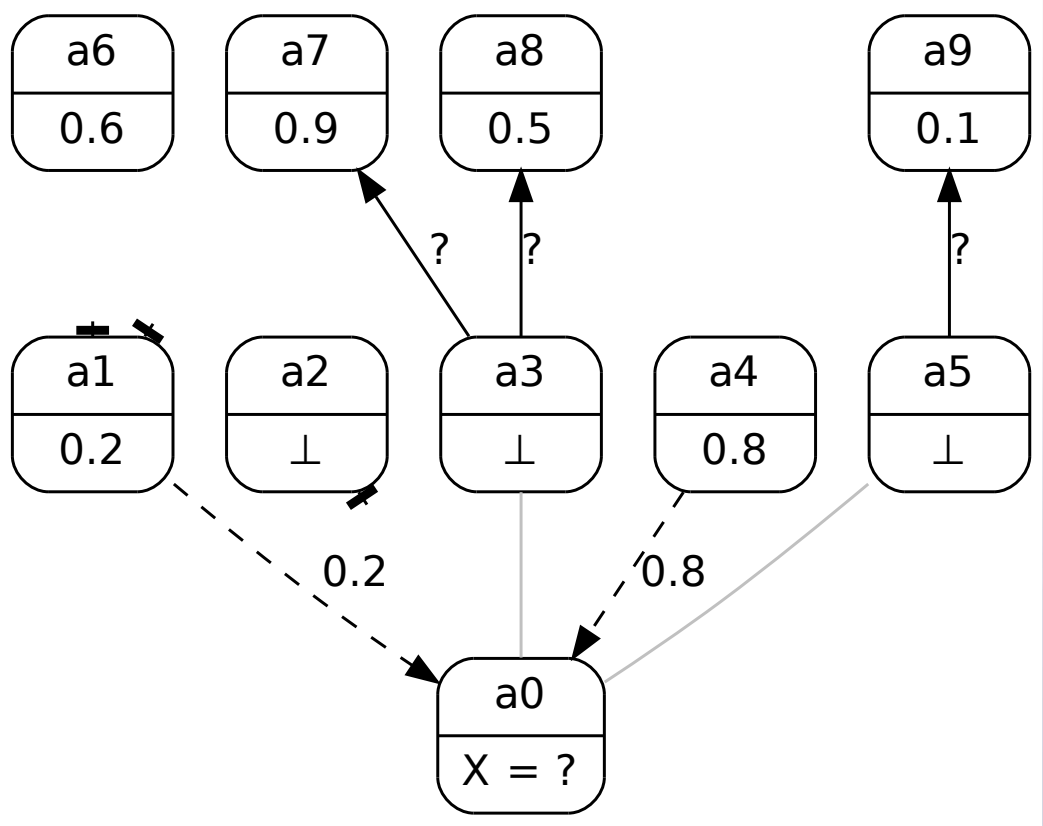


Figure : $k-$Depth Social Scoring Example with $k = 2$

Score propagation example

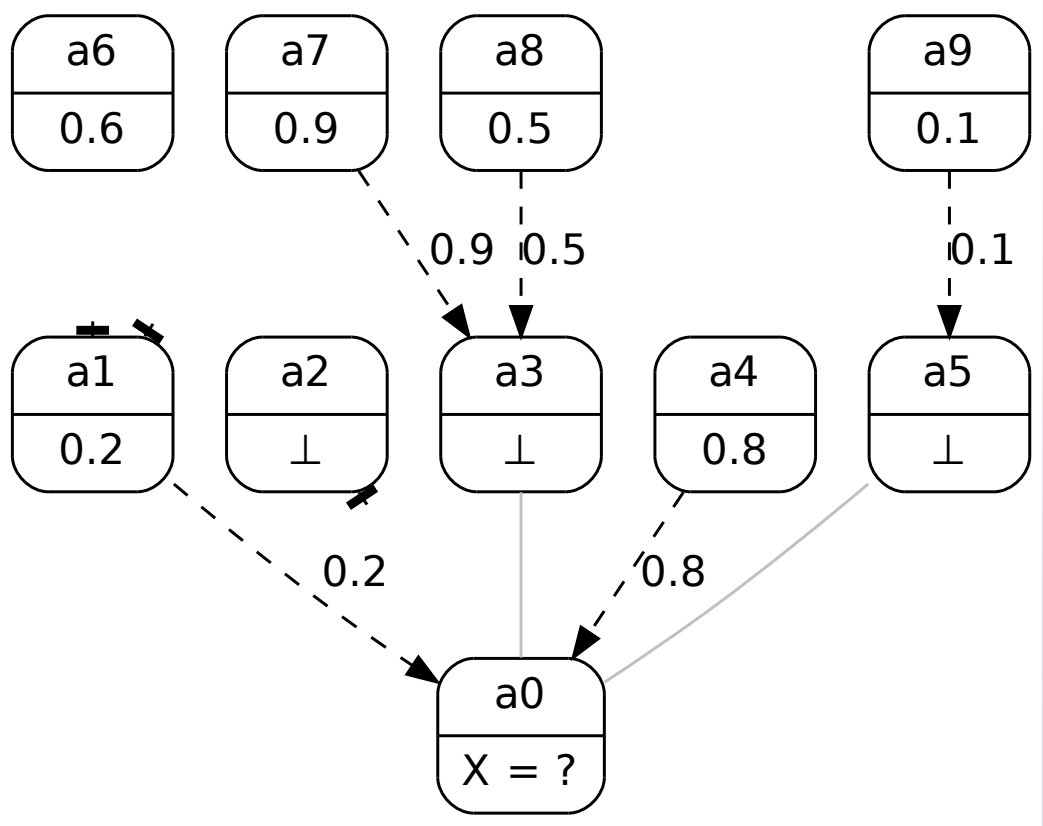


Figure : $k-$Depth Social Scoring Example with $k = 2$

Score propagation example

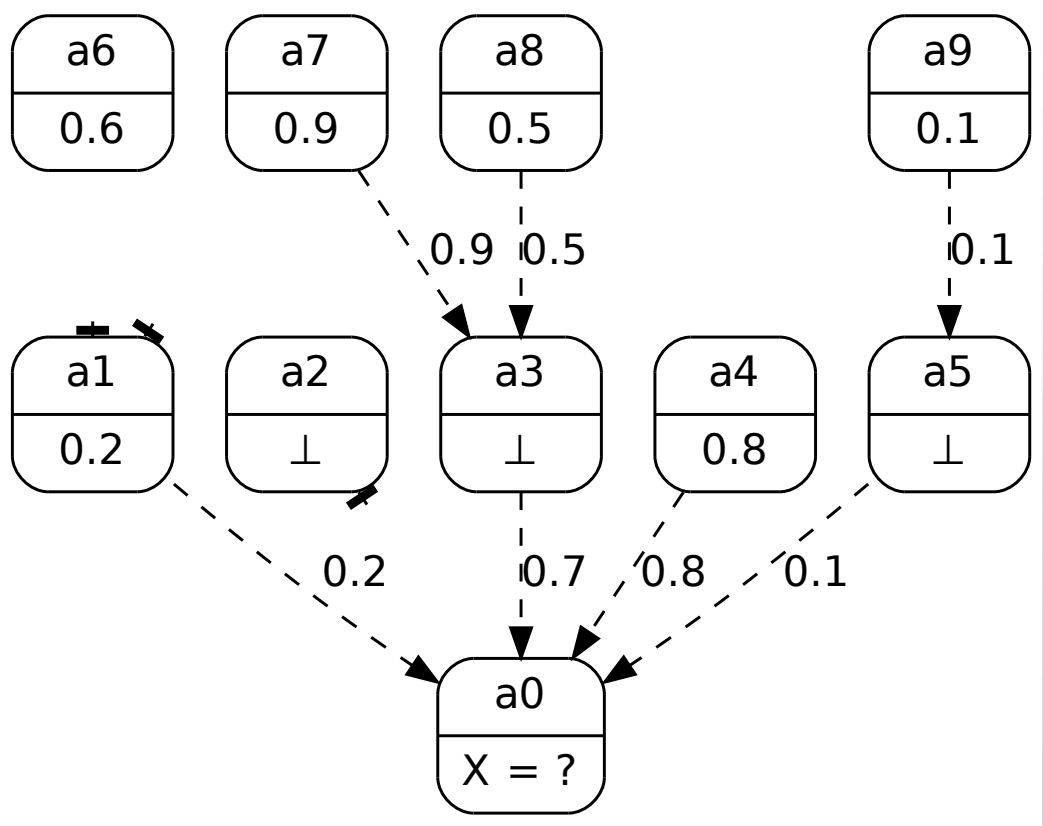


Figure : $k-$Depth Social Scoring Example with $k = 2$

Score propagation example

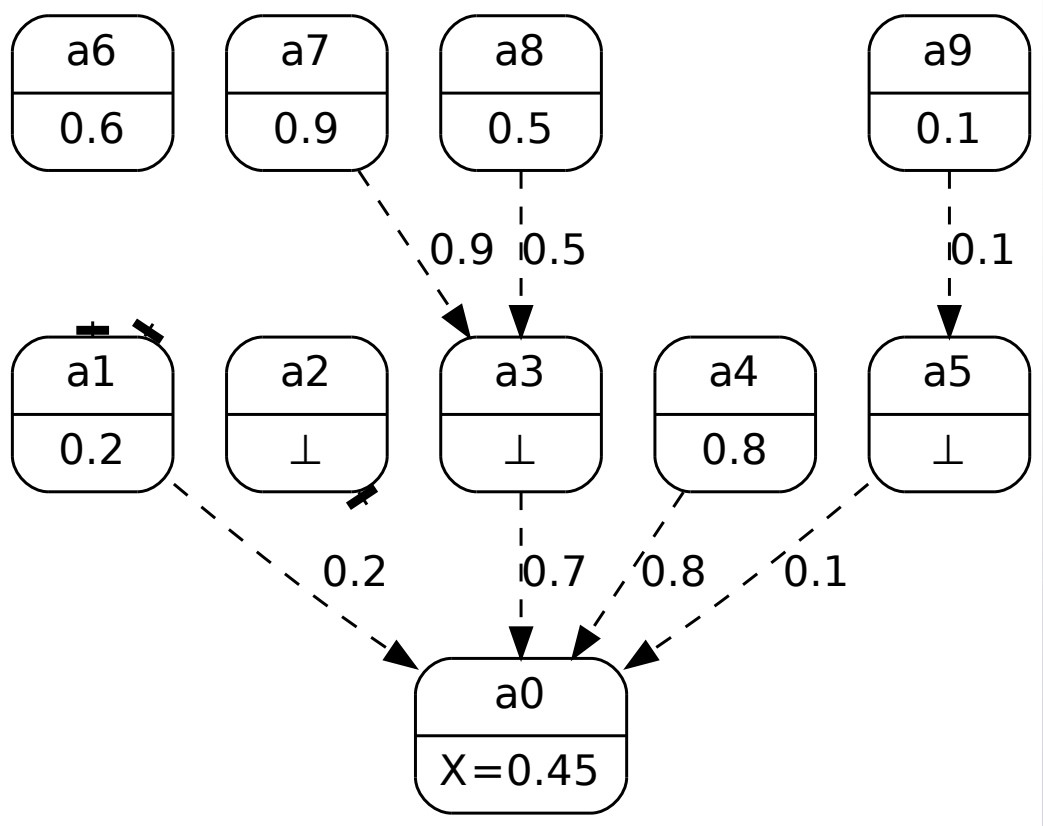


Figure : $k-$Depth Social Scoring Example with $k = 2$

Algorithm

$$\mathcal{F}^k_{a,i,\omega} = \{f \in F_a | s_k(f,i) \neq \perp \wedge \omega_{a,f} \neq 0\} \quad (1)$$

$$s_k(a,i) = \begin{cases} r_{a,i} & \text{if } \exists r_{a,i} \\ \dfrac{\sum_{f \in \mathcal{F}^{k-1}_{a,i,\omega}} \omega_{a,f} \times s_{k-1}(f,i)}{\sum_{f \in \mathcal{F}^{k-1}_{a,i,\omega}} \omega_{a,f}} & \text{if } \nexists r_{a,i} \wedge \mathcal{F}^{k-1}_{a,i,\omega} \neq \emptyset \\ default(a,i) & \text{otherwise} \end{cases} \quad (2)$$

Algorithm

$$\mathcal{F}^k_{a,i,\omega} = \{f \in F_a | s_k(f,i) \neq \perp \wedge \omega_{a,f} \neq 0\} \quad (1)$$

$$s_k(a,i) = \begin{cases} r_{a,i} & \text{if } \exists r_{a,i} \\ \dfrac{\sum_{f \in \mathcal{F}^{k-1}_{a,i,\omega}} \omega_{a,f} \times s_{k-1}(f,i)}{\sum_{f \in \mathcal{F}^{k-1}_{a,i,\omega}} \omega_{a,f}} & \text{if } \nexists r_{a,i} \wedge \mathcal{F}^{k-1}_{a,i,\omega} \neq \emptyset \\ default(a,i) & \text{otherwise} \end{cases} \quad (2)$$

Algorithm

$$\mathcal{F}^k_{a,i,\omega} = \{f \in F_a | s_k(f,i) \neq \perp \wedge \omega_{a,f} \neq 0\} \quad (1)$$

$$s_k(a,i) = \begin{cases} r_{a,i} & \text{if } \exists r_{a,i} \\ \dfrac{\sum_{f \in \mathcal{F}^{k-1}_{a,i,\omega}} \omega_{a,f} \times s_{k-1}(f,i)}{\sum_{f \in \mathcal{F}^{k-1}_{a,i,\omega}} \omega_{a,f}} & \text{if } \nexists r_{a,i} \wedge \mathcal{F}^{k-1}_{a,i,\omega} \neq \emptyset \\ default(a,i) & \text{otherwise} \end{cases} \quad (2)$$

ω coefficient

- Trust
  - $\rightarrow$ Defined by actors on friends
- Correlation (*aka* similarity)
  - $\rightarrow$ Computed by the system between friends
- Confidence
  - $\rightarrow$ On the prediction accuracy
  - $\rightarrow$ Propagated with scores
  - $\rightarrow$ Recomputed by each actor

ω coefficient

- Trust
  - $\rightarrow$ Defined by actors on friends
- Correlation (*aka* similarity)
  - $\rightarrow$ Computed by the system between friends
- Confidence
  - $\rightarrow$ On the prediction accuracy
  - $\rightarrow$ Propagated with scores
  - $\rightarrow$ Recomputed by each actor

ω coefficient

- Trust
  - $\rightarrow$ Defined by actors on friends
- Correlation (*aka* similarity)
  - $\rightarrow$ Computed by the system between friends
- Confidence
  - $\rightarrow$ On the prediction accuracy
  - $\rightarrow$ Propagated with scores
  - $\rightarrow$ Recomputed by each actor

Default score

If no score computed, return (or not) a default score

- Improve coverage on sparse datasets
- Add some randomness

Default score computation:

- Local strategy: $\overline{r_a}$
- Anonymous strategy: $\overline{r_i}$

$$P(default(a, i) \neq \perp) = P_{default}$$

Default score

If no score computed, return (or not) a default score

- Improve coverage on sparse datasets
- Add some randomness

Default score computation:

- Local strategy: $\overline{r_a}$
- Anonymous strategy: $\overline{r_i}$

$$P(default(a, i) \neq \perp) = P_{default}$$

CoTCoDepth

Confident Trust Correlative $k-$Depth Social Scorer

Propagation up to $k = 3$ in the social network:

- CoTCoD3: $default(a, i) = \perp$
- CoTCoD3$_a$: $default(a, i) = \overline{r_a}$
- CoTCoD3$_{ia}$: $default(a, i) = \begin{cases} \overline{r_i} & \text{if } \exists\ \overline{r_i} \\ \overline{r_a} & \text{otherwise} \end{cases}$

Plan

Campaign

Epinions dataset[1] extracted by [Richardson2002]:

- "weakly connected": less than 5 friends
- "fairly connected": 5 to 9 friends
- "highly connected": 10 or more friends

Leave-one-out campaign

- One rating at a time

[1] www.epinions.com

Campaign

Epinions dataset[1] extracted by [Richardson2002]:

- "weakly connected": less than 5 friends
- "fairly connected": 5 to 9 friends
- "highly connected": 10 or more friends

Leave-one-out campaign

- One rating at a time

[1] www.epinions.com

Coverage by actors connectivity

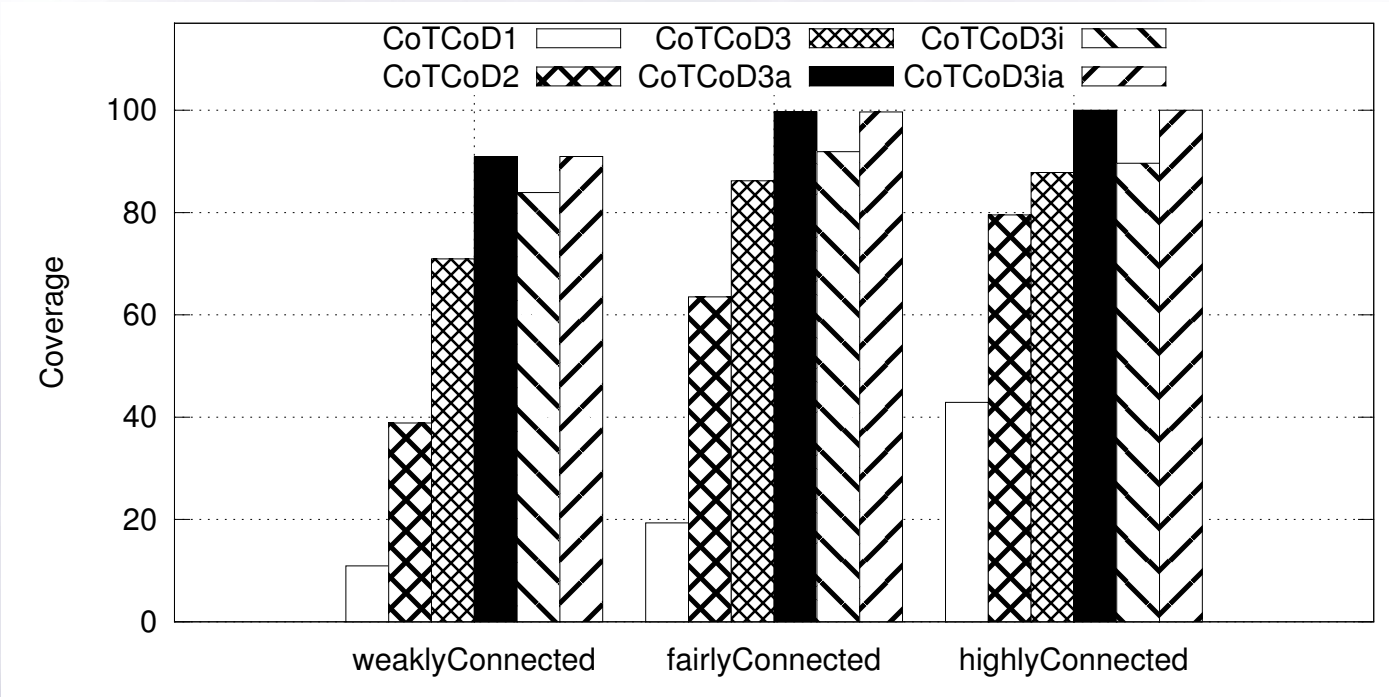


Figure : Coverage of CoTCoDepth scorers

Comparison with existing approaches

Method	Precision	Cov.	F_1	Knowledge
MoleTrust3	0.725	77.25	0.748	extended-local
RandomWalk3	0.682	53.44	0.599	local
CoTCoD3	0.712	77.25	0.741	local
TrustWalker3	0.727	85.99	0.788	local + global
CoTCoD3$_a$	0.723	90.50	0.804	local
CoTCoD3$_{ia}$	0.730	90.56	0.809	local + anonymous

Table : Results for all actors on Epinions

Plan

Conclusion

Local Recommender System

- Only friends share data
- Score propagation
- Users manage their own profiles
- P2P compliant

Features

- Trust and local similarity
- Confidence on scores
- Default scores

Conclusion

Local Recommender System

- Only friends share data
- Score propagation
- Users manage their own profiles
- P2P compliant

Features

- Trust and local similarity
- Confidence on scores
- Default scores

Perspectives

On our recommendation approach

- Content-based
- Extended similarity
- Public profiles (experts)
- ω coefficient depending on the item category

P2P architecture

- Limit network usage (friends subsets)
- Disconnections
- Cache

Perspectives

On our recommendation approach

- Content-based
- Extended similarity
- Public profiles (experts)
- ω coefficient depending on the item category

P2P architecture

- Limit network usage (friends subsets)
- Disconnections
- Cache

Thanks for your attention...

Bibliography

[Jamali2009] Mohsen Jamali and M. Ester.

TrustWalker: a random walk model for combining trust-based and item-based recommendation.

In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 397–406. ACM, 2009.

[Massa2007] Paolo Massa and Paolo Avesani.

Trust-aware recommender systems.

In Proceedings of the 2007 ACM conference on Recommender systems, pages 17–24, New York, New York, USA, 2007. ACM.

[Richardson2002] Matthew Richardson and Pedro Domingos.

Mining knowledge-sharing sites for viral marketing.

In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 61–70, New York, New York, USA, 2002. ACM.

Campaign

Epinions statistics:

- 47 000 users
- 104 000 items
- 586 000 ratings
- 509 000 trust values

Views:

- "weakly connected": 47 % of actors, 22 % of ratings
- "fairly connected": 11 % of actors, 12 % of ratings
- "highly connected": 18 % of actors, 57 % of ratings

Metrics

- Coverage: proportion of predicted ratings regarding all ratings to predict
- Precision: precision metric based on the RMSE (Root Mean Square Error)
- F1-Measure: combination of the coverage and the precision

$$Precision = 1 - \frac{RMSE}{range}$$

$$\text{with } RMSE = \sqrt{\frac{\sum_{n=1}^{N} (p_n - r_n)^2}{N}}$$

$$F_1 = \frac{2 \times Precision \times Coverage}{Precision + Coverage}$$

Metrics

- Coverage: proportion of predicted ratings regarding all ratings to predict
- Precision: precision metric based on the RMSE (Root Mean Square Error)
- F1-Measure: combination of the coverage and the precision

$$Precision = 1 - \frac{RMSE}{range}$$

$$\text{with } RMSE = \sqrt{\frac{\sum_{n=1}^{N} (p_n - r_n)^2}{N}}$$

$$F_1 = \frac{2 \times Precision \times Coverage}{Precision + Coverage}$$

Metrics

- Coverage: proportion of predicted ratings regarding all ratings to predict
- Precision: precision metric based on the RMSE (Root Mean Square Error)
- F1-Measure: combination of the coverage and the precision

$$Precision = 1 - \frac{RMSE}{range}$$

$$\text{with } RMSE = \sqrt{\frac{\sum_{n=1}^{N} (p_n - r_n)^2}{N}}$$

$$F_1 = \frac{2 \times Precision \times Coverage}{Precision + Coverage}$$

Precision

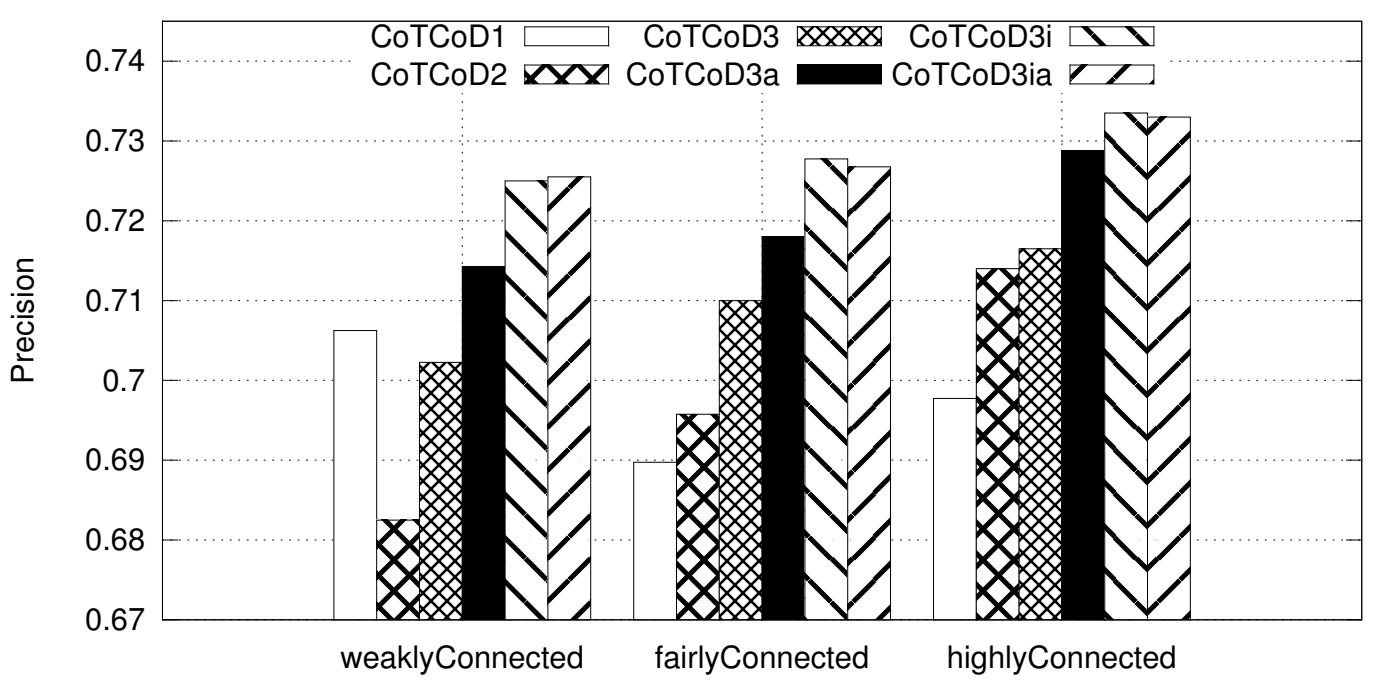


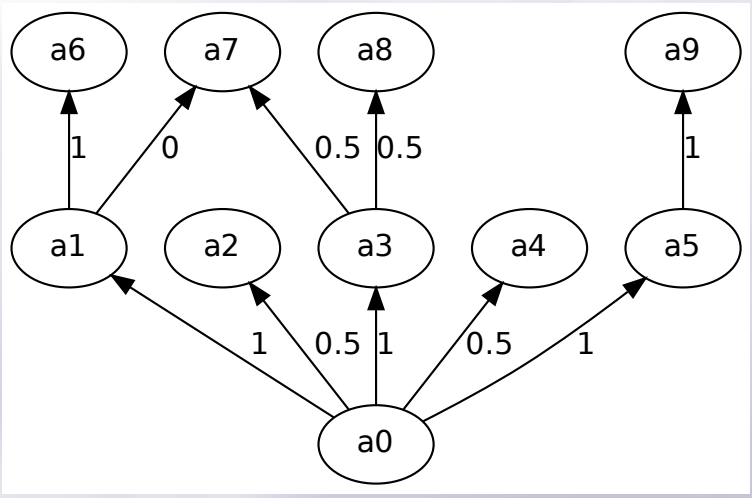
Figure : Precision of CoTCoDepth scorers

Comparison with existing approaches: cold start users

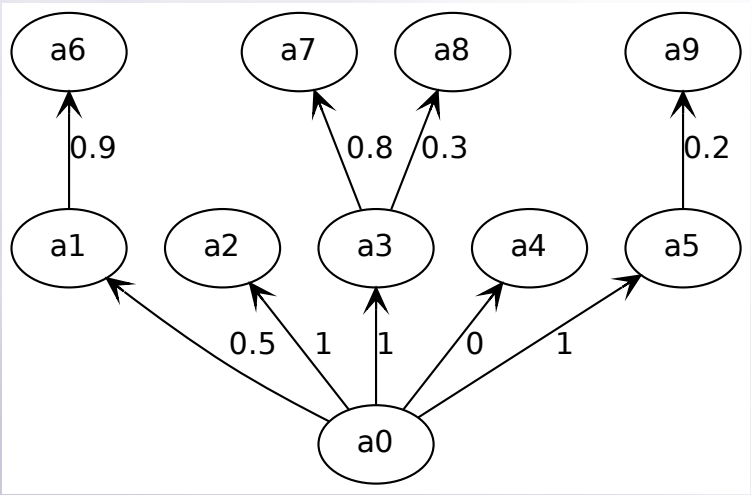
Method	Precision	Cov.	F_1	Knowledge
MoleTrust3	0.708	50.51	0.590	extended-local
RandomWalk3	0.678	37.74	0.485	local
CoTCoD3	0.701	50.51	0.587	local
TrustWalker3	0.678	67.50	0.677	local + global
CoTCoD3$_a$	0.713	65.05	0.681	local
CoTCoD3$_{ia}$	0.724	65.48	0.688	local + anonymous

Table : Results for cold start users on Epinions

ω example



(a) Friends' trust network



(b) Friends' similarity network

Figure : Trust and similarity networks

Default score

$P_{default}$ is the probability to return a defined default score

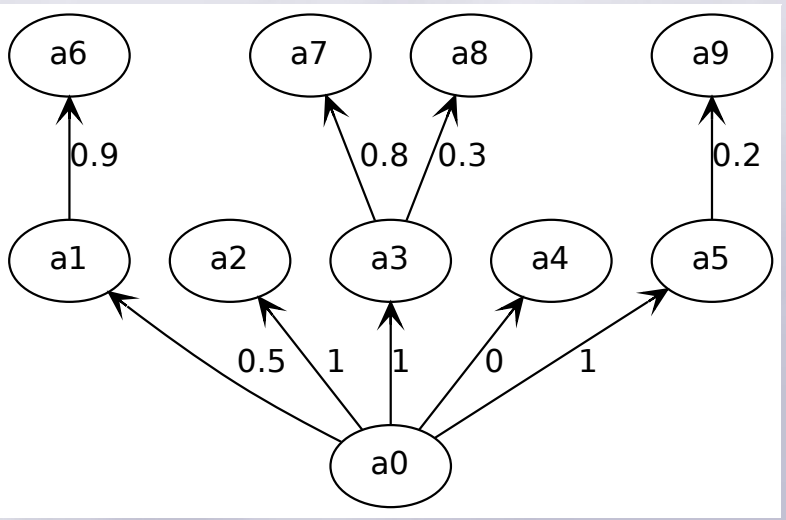
$$P(default(a,i) = \left\{ \begin{array}{c} \overline{r_a} \\ \overline{r_i} \end{array} \right.) = P_{default} \quad (3)$$

$$P(default(a,i) = \perp) = (1 - P_{default}) \quad (4)$$

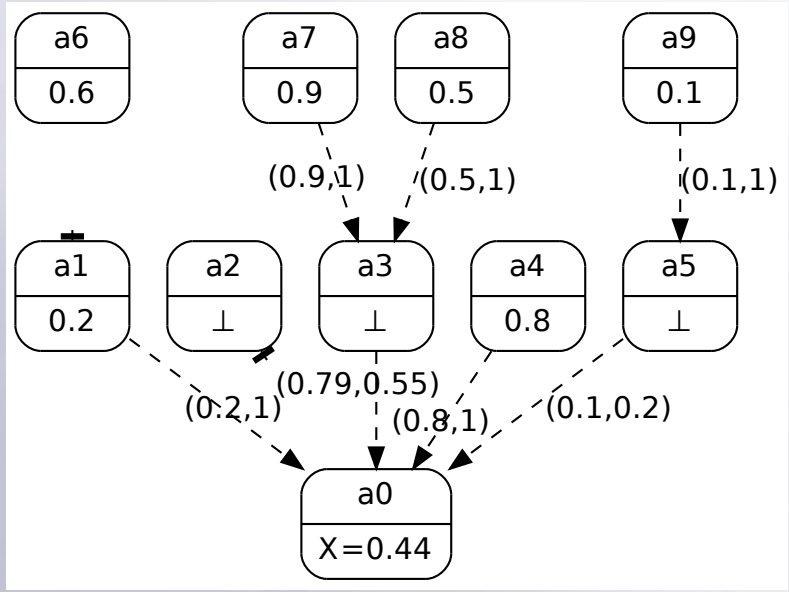
In our experimentations : $P_{default} = 0.02$

Confidence

$$c_{a,i} = \frac{\sum_{f \in \mathcal{F}^k_{a,i,\omega}} \omega_{a,f} \times c_{f,i}}{|\mathcal{F}^k_{a,i,\omega}|} \qquad \omega^{(c)}_{a,f} = \omega_{a,f} \times c_{f,i}$$



(a) Similarity network



(b) Score propagation

Figure : Confidence Example